DUE-DATE SETTING AND PRIORITY SEQUENCING IN A MULTICLASS M/G/1 QUEUE

Lawrence M. Wein

December 1988

Please do not quote, duplicate, or distribute without the author's written permission.
Comments Are Welcome.

DUE-DATE SETTING AND PRIORITY SEQUENCING IN A MULTICLASS M/G/1 QUEUE

Lawrence M. Wein

Sloan School of Management, M.I.T.

Abstract

The problem of simultaneous due-date setting and priority sequencing is analyzed in the setting of a multiclass $M/G/1$ queueing system. The objective is to minimize the weighted average due-date lead time (due-date minus arrival date) of customers subject to a constraint on either the fraction of tardy customers or the average customer tardiness. Several parametric and non-parametric due-date setting policies are proposed that depend on the class of arriving customer, the state of the queueing system at the time of customer arrival, and the sequencing policy (the weighted shortest expected processing time rule) that is used. Simulation results suggest that these policies significantly outperform traditional due-date setting policies and that setting due-dates can have a larger impact on performance than priority sequencing.

December 1988

DUE-DATE SETTING AND PRIORITY SEQUENCING IN A MULTICLASS M/G/1 QUEUE

Lawrence M. Wein

Sloan School of Management, M.I.T.

1. Introduction and Summary

Most of the literature on due-date scheduling problems assume that the due-dates for individual jobs are exogenous. The scheduling problem then becomes one of sequencing the jobs at the various stations in a job shop to optimize some measure of the ability to meet the given due-dates. However, in most firms, the setting of due-dates is negotiable and is the responsibility of the marketing personnel, who have knowledge of the customer's wishes, and the manufacturing personnel, who have knowledge of the shop floor's capability. If the marketing group sets the due-dates oblivious to the shop floor's capability, then the result is often an overloaded shop with a large work-in-process inventory and many jobs past due. On the other hand, if the manufacturing personnel set the due-dates oblivious to the relative importance and urgency of the various jobs, then the customers' wishes will not be satisfactorily addressed. Thus, it is very important for due-dates to be based on the knowledge of the status of the shop floor and the urgency and importance of the various jobs.

In this paper we study two problems of simultaneous due-date setting and priority sequencing in a multiclass $M/G/1$ queueing system. Although this system is an idealized setting, it still captures the dynamic and stochastic elements that are inherent in all job shops. We assume that jobs of each class arrive to the shop according to an independent Poisson process, and the service times for each job class are independent and identically

distributed random variables. The scheduler must assign a due date to each arriving customer and must also dynamically decide in which order to serve the customers in queue. Preemption of the customer in service is not allowed.

Since we assume discretionary due dates, there are two conflicting objectives. Let us define the *due-date lead time* (abbreviated hereafter by DDLT) to be the length of time between a job's arrival to the system and its promised delivery date. The first objective is to set the due-dates as *tight* as possible; that is, set the due-dates to minimize the average DDLT's of jobs. If a firm can reduce their DDLT's, then they can achieve a competitive advantage and will be able to attract more business and/or demand higher prices. However, some job classes may be more important to the firm (due to potential future sales, for example) than others, and thus a more appropriate objective can be found by assigning weights to the various classes and minimizing the weighted average DDLT's.

Once the DDLT's are set, the second and conflicting goal of the shop is to back up their promises and meet the due dates. There are three common measures of the *service level*, or the ability to meet due-dates. One measure is the *lateness* of a job, which is the job's actual completion time minus its due-date. The lateness of a job may be positive or negative, and typical objectives are to minimize the mean and standard deviation of job lateness. The second measure is the *tardiness* of a job, which equals the job's lateness if the lateness is positive, and equals zero otherwise. The typical objective in this case is to minimize the average tardiness of jobs. The final measure is the number of tardy jobs, or equivalently, the fraction of tardy jobs. The typical objective here is to maximize the proportion of jobs that are completed on or before their due-date. In this paper, we will focus on the latter two of these three measures, since they are used more in practice.

It is clear that the two objectives of DDLT minimization and service level maximization are conflicting, since the shorter the DDLT's that a shop quotes, the more difficult it is to achieve a given level of service. Since these two objectives are conflicting, it is perhaps most insightful to state our two scheduling problems in terms of a single objective and a

single constraint. We will refer to a *due-date management policy* as a combination of a due-date setting policy and a priority sequencing policy. Our first problem (denoted by Problem I) is to find a due-date management policy that minimizes the long-run expected weighted average DDLT subject to a constraint on the long-run expected average proportion of tardy jobs. The second problem (Problem II) has the same objective but is instead subject to a constraint on the long run expected average tardiness of jobs.

We will not explicitly solve Problems I and II; the goal of this paper is instead to identify new due-date management policies that outperform conventional due-date management policies appearing in the literature. Due-date management policy A will be considered superior to due-date management policy B in either problem above if both policies satisfy the appropriate constraint with equality, and policy A achieves a lower objective value than policy B. Before stating our results, we will review the relevant literature.

There have been several simulation studies, including Eilon and Chowdhury [9] and Weeks [22], concluding that due-dates based on job content and simple estimates of shop congestion lead to better shop performance that due-dates based solely on job content. The analytical work on this problem include Bertrand [4], who uses a time-phased representation of workload and machine capacity to set workload-dependent due-dates, and Seidmann and Smith [19], who derive a constant due-date assignment policy (that is, the DDLT equals a constant) in a dynamic job shop that minimizes a particular penalty cost.

The two studies that are most closely related to ours are Bookbinder and Noor [5] and Baker and Bertrand [2]. Bookbinder and Noor [5] look at minimizing DDLT for a single machine problem subject to a constraint on the fraction of tardy jobs, and set due-dates based on shop content, job information, and the sequencing policy. However, they assume that the FIFO (first-in first-out) rule is used between batches of jobs. Their due-date setting rule appears to be the only non-parametric rule in the literature; most rules include at least one parameter that must be adjusted (usually via simulation) in order to satisfy some criterion. Baker and Bertrand [2] compare three parametric due-date setting

strategies for a single machine model and a fixed set of jobs. The three rules set a job's DDLT equal to a constant, a constant plus the job's expected processing time, and a constant times the job's expected processing time. (The parameter is the constant in these three cases.) The authors pose the problem of minimizing the average DDLT subject to no jobs being tardy and, under the assumption of known processing times, prove that the first policy is dominated by the other two. We will be incorporating these three policies, none of which depend explicitly on the status of the shop floor, into the simulation experiment in Section 7.

In summary, although there have been many simulation studies and some analytic results, there has been no attempt to set due-dates and sequence jobs in a *unified* manner that will lead to the minimization of DDLT. Although the analyses of Problems I and II appear to be very difficult, an improvement on the existing literature can be made by observing the relationship between DDLT and *cycle time*, where the cycle time of a job is the length of time it spends in the shop. By the definitions of DDLT, cycle time, and lateness, it follows that the DDLT equals the cycle time minus the lateness. Thus if we can find an accurate due-date setting policy (thereby keeping the lateness small), then the desired sequencing policy should aim to minimize the weighted average cycle time in order to achieve the objectives in Problems I and II. It is well known (see, for example, Klimov [17], Harrison [11], and Tcha and Pliska [20]) that the sequencing policy that minimizes the weighted average cycle time in a multiclass $M/G/1$ queue is the weighted shortest expected processing time rule, which is often referred to as the $c\mu$ rule.

As mentioned in Baker and Bertrand [2], if a fixed set of jobs was considered and all processing times were known with certainty, then we could sequence the jobs by the weighted shortest processing time rule and choose due-dates so that the lateness of all jobs equalled zero. This due-date management policy would then minimize the weighted average DDLT objective in Problems I and II and no job tardiness would ever occur. Of course, in a dynamic stochastic environment, one cannot choose due-dates with such

impressive results. But we propose that the $c\mu$ rule be used for sequencing in the $M/G/1$ system, since it would be effective for minimizing the weighted average DDLT of jobs, assuming an accurate due-date setting policy can be found that satisfies the constraints in Problems I and II.

In this paper we propose six due-date setting rules, all of which satisfy the appropriate constraints in Problems I and II; two of these rules are parameterized and the other four rules are non-parameterized. The two parameterized rules apply to both Problems I and II, and, of the four non-parameterized rules, two apply to each problem. We assume that the scheduler in Problems I and II can observe, at the time of each customer arrival, the number of customers of each class in queue (not including service), the class of customer in service (if any), and the length of time that the customer has been in service. The scheduler does not know the times of subsequent arrivals or the service times of customers before their realization. Given the scheduler's knowledge of the queueing system, we define the *conditional sojourn time* of an arriving customer to be the total time the customer spends in the queueing system if the $c\mu$ rule is being employed. The conditional sojourn time for each customer is a random variable that depends on the class of the arriving customer and the state of the system at the time of the customer's arrival. Using standard arguments from the theory of priority queues (see Cobham[6], Kesten and Runnenburg [15], and Conway et al. [7]), we derive the (state-dependent) expected value and Laplace tranform of the conditional sojourn time.

The first parametric rule is based on the mean of the conditional sojourn time distribution, and the second parametric rule is based on the mean and standard deviation of the conditional sojourn time distribution. Two of the non-parametric rules (one for each of Problem I and Problem II) are found by analyzing the tail of the conditional sojourn time distribution.

Unfortunately, the non-parametric rules are difficult to calculate for a general multi-class $M/G/1$ queue. For the simple example described in Section 5 (two customer classes

with different exponential processing times), we use Newton's method to compute the non-parametric due-dates for the higher priority class and, for the lower priority class, use an efficient algorithm recently developed for finding tails of distributions from Laplace transforms by Platzman et al. [18]. For more difficult examples, this method or others, such as Jaegerman [13] or Keilson et al. [14], may be helpful.

Notice that the rules described above do not take into account information about the due-dates that have already been set. We propose two other non-parametric due-date setting rules (one for Problem I and one for Problem II) that use past due-date information to exploit *hot streaks* (a string of fast service times) by the server. The main idea behind these policies is to allow an arriving customer to move ahead of customers of its own class that are in queue (and hence to receive an earlier due-date) as long as these customers will still be expected to depart the system in the desired amount of time.

Using a simulation model of a very simple system, we test seven due-date setting rules (four proposed here and three proposed in [2]) for each of Problems I and II. These rules are used in conjunction with the $c\mu$ rule (which is the shortest expected processing time rule in this example) and various non-parametric due-date sequencing policies (such as the earliest due-date rule) that appear in the literature. As mentioned above, the queueing system has two customer classes that have different exponential processing times.

The primary insight from the simulation study is that proper due-date setting offers a much larger improvement in performance than priority sequencing. The proposed due-date setting policies reduced the mean DDLT by 25-50% in Problem I and 50-68% in Problem II relative to conventional due-date setting policies. It is interesting to point out that the parametric rule based on the first two moments was only slightly more effective at reducing DDLT than the parametric rule based on one moment. Also, there was not a significant difference in performance between the parametric rules and the non-parametric rules; the parametric rules were slightly more effective in Problem I and the non-parametric rules were slightly more effective in Problem II. Thus, the parametric rule based on the

expected value of the conditional sojourn time, which is very easy to calculate for any multiclass $M/G/1$ system, appears to be a very effective due-date setting policy. This has important implications for the more complicated network setting, since it appears to be quite difficult to obtain good estimates of second moments or tails of conditional sojourn time distributions in a multiclass queueing network under various priority schemes. Care must be taken in drawing broad conclusions concerning the relative strength of the four proposed due-date setting policies, since only a single instance of an M/M/1 system has been analyzed numerically.

The simulation results also suggest that when the due-dates proposed here are used, the impact from priority sequencing is minimal. This is intuitively clear, because the proposed due-dates are set in accordance with the $c\mu$ rule, and thus due-date based sequencing policies will not differ greatly from the $c\mu$ rule. However, when using the traditional due-date setting policies (from [2]), priority sequencing has some impact, but not nearly as much as the proposed due-date setting policies. Thus, although there have been many simulation studies (see Baker [1], for example) comparing various priority sequencing heuristics for due-date scheduling problems, it appears that more leverage can be gained by being concerned with the setting of due-dates, not with priority sequencing.

The results of this simulation study closely parallel those of Wein [23], where the problem of simultaneous input control (how to release jobs onto the factory floor) and priority sequencing is analyzed in the setting of a 24-station simulation model of a semiconductor wafer fab. It was found that (1) input control (loosely based on the analysis of Wein [24]-[25]) provided a much larger improvement in performance than did priority sequencing, (2) under proper input control, the effect of priority sequencing was minimal, and (3) under traditional input control, priority sequencing had a moderate impact. The connection between this study and the present one is that due-date setting and input control can both be thought of as *tactical design decisions* that are made at a *higher level* than the priority sequencing decisions. Thoughtful decisions made at this higher level lead to well designed

systems that are much easier to control (in that detailed control issues such as priority sequencing do not need to be a major concern) than poorly designed systems.

This paper is organized as follows. Problems I and II are formulated in Section 2 and the Laplace transform and expected value of the conditional sojourn time are given in Section 3. Two parametric and two non-parametric due-date setting policies are proposed in Section 4. In Section 5, these rules are derived for the case of two customer classes with different exponential processing times. In Section 6, we describe two additional non-parametric due-date setting policies that attempt to exploit hot streaks by the server, and the simulation experiment is presented in Section 7.

2. Two Problem Formulations

We consider a multiclass $M/G/1$ queueing system where jobs of class $k = 1, ..., K$ arrive according to an independent Poisson process with rate λ_k. The service times for job class k are independent and identically distributed random variables with mean μ_k^{-1}, finite variance, general distribution $F_k(t), t \geq 0$, and Laplace transform $F_k^*(s)$. The scheduler must assign a due date $D_{k,t}$ to a class k customer who arrives at time t, and must also dynamically decide in which order to serve the jobs in queue. Thus the DDLT of a class k job that arrives at time t is $D_{k,t} - t$. To repeat, a due-date management policy is a combination of a due-date setting policy and a priority sequencing policy. For $k = 1, ..., K$, let $\bar{D}_k$ be the long-run expected average DDLT of class k jobs. Let $\bar{P}$ be the long-run expected average proportion of jobs that are tardy. Then Problem I is to find a due-date management policy to

$$\text{minimize} \quad \sum_{k=1}^{K} c_k \bar{D}_k \tag{1}$$

$$\text{subject to} \quad \bar{P} \leq \bar{p}, \tag{2}$$

where c_k is a linear cost (or weight) for job class k, and $\bar{p}$ is the desired upper bound on the proportion of jobs that are tardy. For example, many companies define their service goals by desiring to deliver 95% of their jobs on time, in which case $\bar{p} = .05$.

Let $\bar{T}$ be the long-run expected average job tardiness. Then Problem II is to find a due-date management policy to

$$\text{minimize} \ \sum_{k=1}^{K} c_k \bar{D}_k \tag{3}$$

$$\text{subject to} \ \bar{T} \leq \bar{\tau}, \tag{4}$$

where $\bar{\tau}$ is the desired upper bound on the average job tardiness.

The non-parametric rules proposed in this paper can accomodate *class-dependent* service level constraints in Problems I and II. For example, constraint (2) can be replaced by $\bar{P}_k \leq \bar{p}_k$ for $k = 1, ..., K$, where $\bar{P}_k$ is the long-run expected average proportion of class k jobs that are tardy, and $\bar{p}_k$ is the desired upper bound on the proportion of class k jobs that are tardy.

3. The Conditional Sojourn Time

The goal of this section is to derive the Laplace transform and expected value of the conditional sojourn time $S_{k,t}$ for a class k customer who arrives at time t. Recall that the conditional sojourn time of an arriving customer is the total time the customer spends in the system if the $c\mu$ rule is being used, conditioned on the class of arriving customer and the state of the queueing system at the time of arrival. In order to simplify notation, we will suppress the dependence on the arrival time. Without loss of generality, assume the customer classes are ordered so that $c_1\mu_1 \geq ... \geq ...c_K\mu_K$, where c_k is the weight for class k customers in objective function (1) and (3). If a customer arrives to the queueing system at time t, the scheduler can observe the K-dimensional vector $Q(t) = (Q_k(t))$,

where $Q_k(t) = n_k$ is the number of class k customers in queue (not including service) at time t, the class of customer who is currently in service at time t, and the length of time that has elapsed since this customer started service, which is denoted by $a(t) = a$. Let us define F_0 to be the distribution of the residual processing time of the customer currently in service. Then $F_0(t) = 0$ for $t \geq 0$ if the server is idle, and

$$F_0(t) = \frac{F_k(a+t) - F_k(a)}{1 - F_k(a)} \tag{5}$$

if a class k customer is in service, for $k = 1, ..., K$. Let μ_0^{-1} and $F_0^*(s)$ denote the mean and Laplace tranform, respectively, of the residual processing time. It is well known from renewal theory that $\mu_0^{-1} = 0$ if the server is idle, and

$$\frac{1}{\mu_0} = \frac{\int_0^\infty x^2 dF_k(x)}{2\int_0^\infty x dF_k(x)} \tag{6}$$

if class k is in service. Following the notation and reasoning of Chapter 8 of Conway et al.[7], let $\lambda_{ak} = \sum_{i=1}^{k-1} \lambda_i$ be the total arrival rate of jobs with higher priority than class k., let $F_{ak}(t) = \lambda_a^{-1} \sum_{i=1}^{k-1} \lambda_i F_i(t), t \geq 0$ be their composite processing time distribution, and let $F_{ak}^*(s)$ be the associated Laplace transform. Define $G_0^*(s) = F_0^*(s) \prod_{i=1}^{k-1} [F_i^*(s)]^{n_i}$ to be the Laplace transform for the sum of the processing times of the job currently in service plus the processing times of all jobs in queue of higher priority than class k. If we denote the Laplace transform of $S_{k,t}$ by $S_{k,t}^*(s)$, then it follows that for $k = 1, ..., K$,

$$S_{k,t}^*(s) = F_k^*(s)[G_0^*(s + \lambda_{ak} - \lambda_{ak} B_{ak}^*(s))][F_k^*(s + \lambda_{ak} - \lambda_{ak} B_{ak}^*(s))]^{n_k} \tag{7}$$

where $B_{ak}^*(s)$ is the solution to

$$B_{ak}^*(s) = F_{ak}^*(s + \lambda_{ak} - \lambda_{ak} B_{ak}^*(s)). \tag{8}$$

Thus, in order to obtain a closed form solution for $S_{k,t}^*(s)$, one needs to first find the solution $B_{ak}^*(s)$ to equation (8). In cases where a solution can be found, the expected value and standard deviation of the conditional sojourn time, denoted by $E[S_{k,t}]$ and $\sigma[S_{k,t}]$, respectively, can be found by differentiating the Laplace transform $S_{k,t}^*(s)$.

However, a simple expression for $E[S_{k,t}]$ can be found by using the direct expected value procedure of Cobham [6]; see, for example, Dolan [8] or pages 205-207 in Gross and Harris [10]. Let $\rho_k = \lambda_k/\mu_k$ and $\sigma_k = \sum_{i=1}^{k} \rho_i$ for $k = 1, \ldots, K$, where $\sigma_0 = 0$. Then it follows that

$$E[S_{k,t}] = \frac{1}{\mu_k} + \frac{\sum_{i=1}^{k} \frac{n_k}{\mu_k} + \frac{1}{\mu_0}}{1 - \sigma_{k-1}}, \tag{9}$$

where there are n_k class k customers in queue at time t, and where μ_0 is given by (6).

4. Four Due-date Setting Policies

In this section we present four due-date setting policies, all of which satisfy the appropriate constraint (2) or (4). The first two policies are parameterized rules that apply to both Problems I and II. Of the last two policies, which are both parameterized, one applies to Problem I and one applies to Problem II. The first parametric rule assigns the due-date

$$D_{k,t} = t + \alpha E[S_{k,t}], \tag{10}$$

and the second parameterized rule assigns the due-date

$$D_{k,t} = t + E[S_{k,t}] + \beta\sigma[S_{k,t}]. \tag{11}$$

The parameters α and β in (10) and (11) are set (via simulation) in Problems I and II so that the constraints in these problems are satisfied.

The non-parametric rule for Problem I assigns the due-date

$$D_{k,t} = t + p_{k,t}, \tag{12}$$

where

$$\bar{p} = P(S_{k,t} > p_{k,t}). \tag{13}$$

Thus, $p_{k,t}$ is the $(1-\bar{p})$th fractile of the distribution of the random variable $S_{k,t}$. If this due-date setting rule is employed in conjunction with the $c\mu$ rule, then constraint (2) will be automatically satisfied.

Suppose for now that the random variable $S_{k,t}$ has distribution $G_{k,t}$. Then the first non-parametric rule for Problem II assigns the due-date

$$D_{k,t} = t + \tau_{k,t}, \tag{14}$$

where

$$\bar{\tau} = \int_{\tau_{k,t}}^{\infty} (x - \tau_{k,t}) dG_{k,t}(x). \tag{15}$$

Similarly, if this due-date setting rule is employed with the $c\mu$ rule, then constraint (4) will be satisfied.

As mentioned earlier, the due-date setting policies described in (11)-(15) are not easy to calculate for general multiclass $M/G/1$ queues. In the next section we will derive $E[S_{k,t}], \sigma[S_{k,t}], p_{k,t}$, and $\tau_{k,t}$ for a particular simple example.

5. An Example

Suppose there are $K = 2$ customer classes that have exponential processing times with rates μ_1 and μ_2, respectively. Without loss of generality, suppose that $c_1\mu_1 \geq c_2\mu_2$, so that the $c\mu$ rule awards higher priority to class 1 customers. By the memoryless property of the exponential distribution, the residual processing time distribution F_0 is exponential with parameter μ_k if class k customer is in service, for $k = 1, 2$. The state of the system at the time of a customer arrival is adequately described by (n_1, n_2, i_1, i_2), where n_k class k customers are in queue, and i_k equals one if a class k customer is in service, and i_k equals zero otherwise. Notice that $i_1 = i_2 = 0$ implies that $n_1 = n_2 = 0$.

Let us begin by analyzing the conditional sojourn time $S_{1,t}$ of the higher priority customer class. If $i_2 = 0$, then $S_{1,t}$ has an Erlang distribution with shape parameter

n_1+i_1+1 and scale parameter μ_1. If $i_2 = 1$, then $S_{1,t}$ is distributed as the convolution of an Erlang distribution with shape parameter n_1+1 and scale parameter μ_1, and an exponential distribution with parameter μ_2. Thus, when $i_2 = 0$, it follows that $E[S_{1,t}] = \mu_1^{-1}(n_1+i_1+1)$ and $\sigma[S_{1,t}] = \mu_1^{-1}\sqrt{n_1+i_1+1}$. In order to find $p_{1,t}$ in equation (13) when $i_2 = 0$, let $y = \mu_1 p_{1,t}$. Then equation (13) reduces to

$$\bar{p}e^y = \sum_{j=1}^{n_1+i_1+1} \frac{y^{j-1}}{(j-1)!}. \tag{16}$$

When $i_1 = 0$, equation (16) has a closed form solution that leads to $p_{1,t} = \mu_1^{-1}\ln(\bar{p}^{-1})$. When $i_1 = 1$, equation (16) can easily be solved using Newton's method. Similarly, when $i_2 = 0$, equation (15) reduces to

$$\bar{\tau} = \frac{\mu_1^{n_1+i_1+1}e^{-\mu_1\tau_{1,t}}}{(n_1+i_1)!}\left(\frac{(n_1+i_1+1)!}{\mu_1^{n_1+i_1+2}} + \sum_{j=1}^{n_1+i_1}\frac{\tau_{1,t}^n}{\mu_1^{n_1+i_1-j+2}}\left(\frac{(n_1+i_1+1)!}{j!} - \frac{(n_1+i_1)!}{(j-1)!}\right)\right). \tag{17}$$

Once again, a closed form solution $\tau_{1,t} = -\mu_1^{-1}\ln(\mu_1\bar{\tau})$ exists when $i_1 = 0$, and Newton's method can be used when $i_1 > 0$.

When $i_2 = 1$, we have $E[S_{1,t}] = \mu_1^{-1}(n_1+1)+\mu_2^{-1}$ and $\sigma[S_{1,t}] = \sqrt{\mu_1^{-2}(n_1+1)+\mu_2^{-2}}$. When $i_2 = 1$, equation (13) reduces to

$$1-\bar{p} = \mu_1^{n_1+1}\mu_2 e^{-\mu_2 p_{1,t}}\left(\sum_{j=1}^{n_1}(-1)^{n_1+j}\frac{p_{1,t}^j e^{(\mu_2-\mu_1)p_{1,t}}}{(\mu_2-\mu_1)^{n_1-j+1}j!} + (-1)^{n_1}\frac{(e^{(\mu_2-\mu_1)p_{1,t}}-1)}{(\mu_2-\mu_1)^{n_1+1}}\right). \tag{18}$$

and equation (15) reduces to

$$\begin{aligned}\bar{\tau} =\mu_1^{n_1+1}\mu_2\Bigg(&\sum_{j=1}^{n_1+1}\frac{(-1)^{n_1+j}e^{-\mu_1\tau_{1,t}}}{(\mu_2-\mu_1)^{n_1-j+1}j!}\left(\sum_{l=1}^{j+1}\frac{\tau_{1,t}^l}{\mu_1^{j-l+2}}\left(\frac{(j+1)!}{l!} - \frac{j!}{(l-1)!}\right) + \frac{(j+1)!}{\mu_1^{j+2}}\right)\\ &+\frac{(-1)^{n_1}}{(\mu_2-\mu_1)^{n_1+1}}\left(\frac{e^{-\mu_1\tau_{1,t}}}{\mu_1^2} - \frac{e^{-\mu_2\tau_{1,t}}}{\mu_2^2}\right)\Bigg),\end{aligned} \tag{19}$$

which can both be solved by Newton's method.

The analysis of the conditional sojourn time $S_{2,t}$ of the lower priority class is more difficult and requires the use of the Laplace transform $S_{k,t}^*$. Since $F_{a2}^*(s) = F_1^*(s) =$

$\mu_1/(s+\mu_1)$, the solution $B^*_{a2}(s)$ to equation (8) is (see, for example, page 215 of Kleinrock [16])

$$B^*_{a2}(s) = \frac{\mu_1 + \lambda_1 + s - \sqrt{(\mu_1+\lambda_1+s)^2 - 4\mu_1\lambda_1}}{2\lambda_1}. \tag{20}$$

By equation (7) we have

$$S^*_{2,t}(s) = \left(\frac{\mu_2}{s+\mu_2}\right)\left(\frac{2\mu_1}{\mu_1+\lambda_1+s+\sqrt{(\mu_1+\lambda_1+s)^2-4\mu_1\lambda_1}}\right)^{n_1+i_1}$$
$$\left(\frac{2\mu_2}{2\mu_2-\mu_1+\lambda_1+s+\sqrt{(\mu_1+\lambda_1+s)^2-4\mu_1\lambda_1}}\right)^{n_2+i_2}, \tag{21}$$

where, for $k=1,2$, $i_k = 1$ if a class k customer is in service at time t, and equals zero otherwise. Readers may verify that $-S^{*'}_{2,t}(0)$ yields the value of $E[S_{2,t}]$ given in equation (9), and that differentiating (21) twice gives

$$\begin{aligned} E[S^2_{2,t}] = S^{*''}_{2,t}(0) = & \frac{2}{\mu_2^2} + (n_1+i_1)\left(\frac{2}{\mu_2(\mu_1-\lambda_1)} + \frac{2\lambda_1}{(\mu_1-\lambda_1)^3}\right) \\ & + (n_2+i_2)\left(\frac{2\mu_1}{\mu_2^2(\mu_1-\lambda_1)} + \frac{2\lambda_1\mu_1}{\mu_2(\mu_1-\lambda_1)^3}\right) \\ & + \frac{1}{(\mu_1-\lambda_1)^2}\Bigg((n_1+i_1)(n_1+i_1+1) + \frac{\mu_1^2}{\mu_2^2}(n_2+i_2)(n_2+i_2+1) \\ & + \frac{2\mu_1}{\mu_2}(n_1+i_1)(n_2+i_2)\Bigg). \end{aligned} \tag{22}$$

Thus,

$$\begin{aligned} \sigma[S_{2,t}] &= \sqrt{E[S^2_{2,t}] - E[S_{2,t}]^2} \\ &= \sqrt{\frac{1}{\mu_2^2} + (n_1+i_1)\left(\frac{\lambda_1+\mu_1}{(\mu_1-\lambda_1)^3}\right) + (n_2+i_2)\left(\frac{\mu_1^3 - \lambda_1\mu_1^2 + 2\lambda_1\mu_1\mu_2}{\mu_2^2(\mu_1-\lambda_1)^3}\right)}. \end{aligned} \tag{23}$$

Since $E[S_{k,t}]$ does not equal $\sigma[S_{k,t}]$ for $k=1,2$, due-date setting policies (10) and (11) should yield different results for our example.

In order to calculate $p_{2,t}$ and $\tau_{2,t}$ in equations (13) and (15), we used an approximation algorithm recently developed by Platzman et al. [18] for computing tail probabilities

from transforms. For a prespecified accuracy parameter ΔA and prespecified precision parameter ΔP, they showed that the tail probability TP defined by

$$TP = \frac{U - A + \Delta A}{U + 2\Delta A} + \sum_{n=1}^{N} \frac{\alpha^{n^2}}{\pi n} \mathrm{im}\{(\beta^n - \gamma^n) S_{2,t}^*(j\omega n)\} \tag{24}$$

satisfies

$$P(S_{2,t} \geq A + \Delta A) - \Delta P \leq TP \leq P(S_{2,t} > A - \Delta A) + \Delta P \tag{25}$$

as long as $P(S_{2,t} \leq U) << \Delta P$. Here,

$$\begin{aligned} &j = \sqrt{-1}, \quad k = \ln(\frac{2}{\Delta P}), \quad D = \frac{\Delta A}{\sqrt{2k}}, \quad \omega = \frac{2\pi}{U + 2\Delta A}, \\ &N = \lceil \frac{2k}{\Delta A \omega} \rceil, \quad \alpha = e^{\frac{-D^2\omega^2}{2}}, \ \beta = e^{j(U+\Delta A)\omega}, \quad \text{and} \quad \gamma = e^{jA\omega}. \end{aligned} \tag{26}$$

By equation (25), it can be seen that equation (24) calculates the appropriate tail probability $P(S_{2,t} > A)$ given A, whereas we need to find the value of A such that the tail probability equals the desired value of $\bar{p}$ in equation (13). However, imbedding equation (24) within a simple search algorithm allows us to calculate $p_{2,t}$. The search algorithm, given previously calculated values of A_{i-1}, TP_{i-1}, A_i, and TP_i, calculates a new value of A, denoted by A_{i+1}, by

$$A_{i+1} = A_{i-1} - (\frac{TP_{i-1} - \bar{p}}{TP_{i-1} - TP_i})(A_{i-1} - A_i). \tag{27}$$

Equation (24) is then used to calculate TP_{i+1} given A_{i+1}. The algorithm is stopped with a solution $p_{2,t} = A_i$ when

$$|TP_i - \bar{p}| \leq \epsilon, \tag{28}$$

where ϵ is a small specified value.

Platzman et al. [18] showed more generally that, for any integrable function $g(S_{2,t})$,

$$TP^* = C_0 + 2\sum_{n=1}^{N} \alpha^{n^2} \mathrm{real}\{C_n S_{2,t}^*(j\omega n)\} \tag{29}$$

always lies in the range

$$\mathrm{conv}\{E[g(S_{2,t}+a)|0 \le S_{2,t} \le U] + b\Delta P : |a| \le \Delta A, |b| \le \mathrm{diam}(g(I))\}, \tag{30}$$

where $\mathrm{diam}(g(I)) = \max\{g(x) - g(x') : x, x' \le I\}$, where the Fourier series coefficients C_n for $n = 0, 1, ..., N$ appearing in equation (29) are given by

$$C_n = \frac{1}{U + 2\Delta A} \int_{-\Delta A}^{U+\Delta A} e^{jn\omega y} g(y) dy. \tag{31}$$

By specializing the function g to

$$g_A(y) = \begin{cases} y - A, & \text{if } y > A, \\ 0, & \text{otherwise,} \end{cases} \tag{32}$$

we can obtain an approximate value of $\tau_{2,t}$ with a similar algorithm that was used to find $p_{2,t}$: equation (29) is simply used in place of equation (24) to find TP_i^* given A_i, and TP^* takes the place of TP in equation (27). For our special case of the function g in (32), the Fourier coefficients are given by

$$C_0 = \frac{U^2 + A^2 + (\Delta A)^2 + 2U\Delta A - 2A\Delta A - 2AU}{2(U + 2\Delta A)} \tag{33}$$

and, for $n = 1, ..., N$,

$$C_n = e^{jn\omega(U+\Delta A)}\Big(\frac{U+\Delta A}{2\pi jn} + \frac{1}{2\pi\omega n^2}\Big) - e^{jn\omega A}\big(\frac{A}{2\pi jn} + \frac{1}{2\pi\omega n^2}\big), \tag{34}$$

where ω is given in (26).

6. Exploiting Hot Streaks

Although the due date $D_{k,t}$ for a class k customer arriving at time t proposed in Section 4 depend on the class k of arriving customer and on the state of the queueing system at time t, it does not depend on the due dates of customers who are in queue at

time t. It is reasonable to presume that improved due-date setting policies for Problems I and II could be found by allowing $D_{k,t}$ to depend on past due-date information.

In this section, we describe two non-parametric due-date setting policies (one for Problem I and one for Problem II) that use past due-date information to exploit hot streaks (a sequence of fast service times) by the server. The situation we are attempting to exploit is the following: suppose a customer arrives at time t and, just prior to time t, the server has completed a sequence of services that were faster than expected. (In the case where machine breakdown and repair are incorporated into the service time distributions (see, for example, Harrison [12]), such a hot streak can occur when there has not been a machine breakdown for an unusually long time, or when a machine is repaired much quicker than expected.) Then there may be customers in queue who have particularly slack due-dates, because their due-dates were set before the start of a hot streak. Indeed, there may be enough slack in these due-dates so that an arriving customer can move ahead of these customers in queue without endangering these customers with tardiness.

With this situation in mind, we define the following non-parametric due-date setting policy for Problem I. This policy only allows an arriving customer to move ahead of customers of its own class. The corresponding sequencing policy still ranks the customer classes by the $c\mu$ rule, but now customers are ranked *within* each class by the earliest due-date, not by the earliest arrival date. This policy first computes new due-dates for each customer in the queue that is of the same class as the arriving customer (say, class k). These due-dates are computed only for the purpose of assigning a due-date to the arriving customer; the customers in queue still retain the original due-date that was assigned to them at the time of their arrival. The new due-date for a customer of class k in queue is again computed according to (12)-(13), but is now computed as *if* the new customer has just arrived and observes n_k customers of class k in queue, where n_k equals $m_k + 1$ and m_k equals the number of class k customers in queue that have an earlier due-date than this customer; the extra one in addition to m_k accounts for the arriving customer.

Thus the new due-date is based on a revised conditional sojourn time that assumes that the arriving customer moves ahead of the customer in queue. If the new due-date of the customer in queue is earlier than the customers original due-date, then there is enough slack in the customer's original due-date to allow the arriving customer to move ahead of this customer in queue; in this case, we say that the customer in queue can *accomodate* the arriving customer.

Under our proposed policy, an arriving customer, rather than joining the end of the queue of class k customers (who are ordered according to the earliest due-date criterion), instead passes customers in its queue (starting from the back of the queue) until he/she meets a customer in queue who cannot accomodate him/her. The arriving job's due date is then calculated by (12)-(13), ignoring all the customers of its class that it has passed in queue.

The corresponding non-parametric rule for Problem II is identical to this rule, except that equations (14) and (15) are used in place of (12) and (13) to calculate all old and new due-dates. If these two due-date setting policies are used in conjunction with the $c\mu$ rule, where jobs are served within each class by the earliest due-date criterion, then constraints (2) and (4), respectively, should be satisfied.

This idea of exploiting server hot streaks can be extended so that arriving customers may pass ahead of customers in queue that are of a higher class, not just of the same class. In this case, the corresponding sequencing policy would be according to the earliest due-date criterion, regardless of the class of customer, although equations (12)-(15) would still be used to set new and old due-dates. However, the conditional sojourn time for a customer in queue would need to take into account all customers in queue who have earlier due-dates than this customer. Such a policy cannot be guaranteed to satisfy constraints (2) or (4), since the $c\mu$ rule would not be used. Due to the rather cumbersome nature of these policies and due to the modest inprovement achieved in the simulation study of the next section by the two rules described earlier in this section, this idea has not been

pursued any further.

7. A Simulation Study

Using the example in Section 5, a simulation study was undertaken to compare the performance of the due-date setting policies described in Sections 4 and 6 against conventional due-date setting policies. The example queueing system has two customer classes with Poisson arrival rates $\lambda_1 = .4$ and $\lambda_2 = .2$. The two classes have exponential service times with rates $\mu_1 = 1$ and $\mu_2 = .5$. Thus, $\rho_1 = \rho_2 = .4$ and the server utilization is $\rho = .8$. The weights for the two customer classes are $c_1 = c_2 = 1$ (that is, the objective in Problems I and II is to minimize the long-run expected average DDLT of jobs), and so the $c\mu$ rule gives higher priority to class 1 jobs. The service levels for problems I and II were set at $\bar{p} = .05$ (that is, 5% tardy jobs) and $\bar{\tau} = 0.5$.

For each of Problems I and II, seven due-date setting policies and five priority sequencing policies were tested; thus, 35 due-date management policies were tested for each problem. For each due-date management policy tested on each problem, 20 independent runs were made, each consisting of 5000 customer completions. Each simulation run started with an empty system and had no initialization period. Five parametric due-date setting policies, the first three of which are from [2], were tested on both problems: a constant policy (referred to as CONSTANT in Tables I and II), where $D_{k,t} = t + c$ for some parameter c; a slack policy (SLACK), where $D_{k,t} = t + \mu_k^{-1} + c$; proportional (PROP), where $D_{k,t} = t + c\mu_k^{-1}$; the policy described in equation (10), which will be referred to as $E[S_{k,t}]$; and policy (11), which will be referred to as $\sigma[S_{k,t}]$. For these policies, the associated parameter was set so that the resulting average fraction of tardy jobs $\bar{P}$ satisfied $\bar{P} \in [.05 \pm .0005]$ and the resulting average job tardiness $\bar{T}$ satisfied $\bar{T} \in [.5 \pm .005]$.

In addition, two non-parametric due-date setting rules were tested on each problem.

The policy described in (12)-(13), which will be referred to as NONPAR I, was tested on Problem I, and policy (14)-(15), which will be referred to as NONPAR II, was tested on Problem II. Also, the two corresponding policies described in Section 6, which will be referred to as HOT I and HOT II, were tested on Problems I and II, respectively. For these four policies, the accuracy parameter ΔA and precision parameter ΔP in equation (25) were set equal to .01 and .005, respectively. Also, the parameter ϵ appearing in equation (28) was set equal to .0005 for NONPAR I and HOT I, and was set equal to .005 for NONPAR II and HOT II. The upper bound parameter U appearing in (24) was set equal to twenty times the expected value of the conditional sojourn time.

Only non-parametric priority sequencing rules were considered in our study; readers are referred to Vepsalainen and Morton [21] for recent work in parameterized rules. The five priority sequencing policies are: the shortest expected processing time rule (SPT), where class 1 jobs get priority over class 2 jobs; the earliest due date rule (EDD), where priority is given to the job with the earliest due-date; the minimum slack rule (SLACK), which gives priority to the job with the smallest slack, where a job's slack is its due-date minus its expected processing time minus the current time; a critical ratio rule (S/EPT), where priority is given to the job with the smallest ratio of its slack divided by its expected processing time; and the modified due-date (MDD) policy of Baker and Bertrand [3], which gives priority to the job with the earliest modified due-date, where a job's modified due-date is the maximum of its due-date and its earliest expected completion time (that is, the current time plus its expected processing time).

For the SPT rule, we need to specify the manner in which customers are ordered *within* each class. Two possibilities are to order them by the earliest arrival date or by the earliest due-date; the two subsequent sequencing policies are denoted by SPT(FIFO) and SPT(EDD), respectively. Under the three traditional due-date setting policies, these two sequencing rules are identical, since the due-dates do not depend on the state of the queueing system. As mentioned in Section 6, the HOT I and HOT II due-date setting rules

need to be run with SPT(EDD) in order to satisfy the necessary constraint. However, the remaining proposed due-date setting policies were tested in conjunction with both the SPT(FIFO) and SPT(EDD) sequencing policies.

The results of the simulation study are summarized in Table I (for Problem I) and Table II (for Problem II). In both tables, each row corresponds to a due-date management policy, and the average DDLT of jobs is given for each row, along with a 95% confidence interval. In addition, the average fraction of tardy jobs is stated in Table I, and the average job tardiness is stated in Table II, both with 95% confidence intervals.

As can be seen in Table I, the four due-date setting policies easily outperformed the three traditional due-date setting policies. Also, the due-date setting policies have a much bigger impact on performance than do the priority sequencing policies. As for our four proposed due-date setting policies, the two non-parametric rules slightly outperformed the two parametric rules. The HOT I policy was the best due-date setting policy, slightly outperforming the NONPAR I policy. Since the ability to exploit hot streaks should increase with the variability of the processing times, and since only a minor improvement was obtained with exponential processing times, it would appear that the HOT I policy will not often lead to a significant improvement over the NONPAR I policy. Also, the $\sigma[S_{k,t}]$ policy achieved a minor reduction in DDLT compared to $E[S_{k,t}]$, suggesting that the second moment of the conditional sojourn time improves performance, but not dramatically.

Notice that the service level target of $\bar{p} = .05$ was well within the 95% confidence intervals of the proportion of tardy jobs observed under the (NONPAR I,SPT) and (HOT I,SPT) due-date management policies; thus the algorithms developed in Sections 5 and 6 for the non-parametric due-date setting policies are shown to be reliable.

There was a negligible difference in performance among the priority sequencing policies when they were used in conjunction with one of the four due-date setting policies proposed for Problem I. This is because the due-dates were set in accordance with the SPT rule, and thus the SPT rule and the due-date based rules prioritized jobs in a similar manner.

DUE-DATE SETTING POLICY	PRIORITY SEQUENCING POLICY	MEAN DDLT	PERCENTAGE TARDY JOBS
CONSTANT	SPT	23.5(±.00)	5.02(±.63)
CONSTANT	EDD	22.3(±.00)	4.99(±.96)
CONSTANT	SLACK	22.43(±.00)	5.00(±.96)
CONSTANT	S/EPT	23.6(±.00)	4.98(±.97)
CONSTANT	MDD	21.7(±.00)	4.99(±.92)
SLACK	SPT	22.88(±.00)	5.02(±.63)
SLACK	EDD	22.43(±.00)	5.01(±.97)
SLACK	SLACK	22.23(±.00)	5.01(±.96)
SLACK	S/EPT	23.23(±.00)	5.05(±.93)
SLACK	MDD	21.73(±.00)	5.00(±.97)
PROP	SPT	16.4(±.03)	4.99(±.60)
PROP	EDD	19.13(±.03)	5.02(±1.04)
PROP	SLACK	19.33(±.03)	4.98(±1.02)
PROP	S/EPT	20.6(±.04)	5.04(±.94)
PROP	MDD	18.7(±.03)	4.95(±1.00)
$E[S_{k,t}]$	SPT(FIFO)	13.9(±.51)	4.99(±.18)
$E[S_{k,t}]$	SPT(EDD)	13.9(±.53)	5.01(±.18)
$E[S_{k,t}]$	EDD	13.8(±.53)	4.97(±.16)
$E[S_{k,t}]$	SLACK	13.8(±.51)	4.95(±.18)
$E[S_{k,t}]$	S/EPT	14.1(±.51)	5.00(±.17)
$E[S_{k,t}]$	MDD	13.8(±.53)	5.01(±.18)
$\sigma[S_{k,t}]$	SPT(FIFO)	12.9(±.36)	4.95(±.23)
$\sigma[S_{k,t}]$	SPT(EDD)	12.4(±.34)	5.05(±.26)
$\sigma[S_{k,t}]$	EDD	13.0(±.38)	4.98(±.28)
$\sigma[S_{k,t}]$	SLACK	13.0(±.38)	5.02(±.26)
$\sigma[S_{k,t}]$	S/EPT	13.0(±.38)	5.01(±.28)
$\sigma[S_{k,t}]$	MDD	12.5(±.40)	5.04(±.30)
NONPAR 1	SPT(FIFO)	12.4(±.33)	4.88(±.24)
NONPAR 1	SPT(EDD)	12.4(±.33)	4.75(±.25)
NONPAR 1	EDD	12.5(±.36)	4.89(±.35)
NONPAR 1	SLACK	12.6(±.36)	4.94(±.35)
NONPAR 1	S/EPT	12.3(±.33)	5.97(±.29)
NONPAR 1	MDD	12.3(±.33)	4.75(±.25)
HOT 1	SPT(EDD)	11.9(±.31)	5.09(±.27)
HOT 1	EDD	12.0(±.33)	5.20(±.36)
HOT 1	SLACK	12.0(±.34)	5.23(±.37)
HOT 1	S/EPT	12.0(±.33)	5.69(±.37)
HOT 1	MDD	11.9(±.33)	5.04(±.35)

TABLE I. SIMULATION RESULTS FOR PROBLEM I.

DUE-DATE SETTING POLICY	PRIORITY SEQUENCING POLICY	MEAN DDLT	MEAN JOB TARDINESS
CONSTANT	SPT	27.0(±.00)	.506(±.124)
CONSTANT	EDD	19.7(±.00)	.506(±.149)
CONSTANT	SLACK	20.0(±.00)	.497(±.150)
CONSTANT	S/EPT	19.8(±.00)	.492(±.137)
CONSTANT	MDD	19.1(±.00)	.497(±.138)
SLACK	SPT	26.33(±.00)	.506(±.124)
SLACK	EDD	19.83(±.00)	.502(±.151)
SLACK	SLACK	19.63(±.00)	.509(±.150)
SLACK	S/EPT	19.7(±.00)	.493(±.137)
SLACK	MDD	19.1(±.00)	.501(±.142)
PROP	SPT	18.0(±.03)	.509(±.125)
PROP	EDD	17.2(±.03)	.493(±.167)
PROP	SLACK	17.33(±.03)	.508(±.154)
PROP	S/EPT	17.3(±.03)	.501(±.144)
PROP	MDD	16.5(±.03)	.499(±.145)
$E[S_{k,t}]$	SPT(FIFO)	8.91(±.33)	.504(±.035)
$E[S_{k,t}]$	SPT(EDD)	8.80(±.34)	.505(±.038)
$E[S_{k,t}]$	EDD	8.60(±.36)	.504(±.034)
$E[S_{k,t}]$	SLACK	8.61(±.34)	.497(±.034)
$E[S_{k,t}]$	S/EPT	10.05(±.41)	.492(±.070)
$E[S_{k,t}]$	MDD	8.75(±.34)	.505(±.038)
$\sigma[S_{k,t}]$	SPT(FIFO)	8.86(±.28)	.498(±.044)
$\sigma[S_{k,t}]$	SPT(EDD)	8.79(±.28)	.502(±.046)
$\sigma[S_{k,t}]$	EDD	8.62(±.30)	.500(±..045)
$\sigma[S_{k,t}]$	SLACK	8.66(±.30)	.501(±.045)
$\sigma[S_{k,t}]$	S/EPT	8.71(±.33)	.507(±.043)
$\sigma[S_{k,t}]$	MDD	8.50(±.32)	.502(±.042)
NONPAR I	SPT(FIFO)	9.50(±.34)	.520(±.029)
NONPAR I	SPT(EDD)	9.53(±.35)	.510(±.031)
NONPAR I	EDD	9.58(±.37)	.489(±.019)
NONPAR I	SLACK	9.59(±.36)	.492(±.017)
NONPAR I	S/EPT	9.48(±.34)	.733(±.081)
NONPAR I	MDD	9.48(±.35)	.510(±.031)
HOT I	SPT(EDD)	9.01(±.32)	.532(±.032)
HOT I	EDD	9.09(±.34)	.514(±.029)
HOT I	SLACK	9.10(±.34)	.517(±.030)
HOT I	S/EPT	9.05(±.33)	.522(±.028)
HOT I	MDD	9.02(±.33)	.511(±.028)

Table II SIMULATION RESULTS FOR PROBLEM II.

The only exception is the S/EPT sequencing rule, which did not perform well with the proposed due-date setting policies. This is not surprising, however, since this sequencing rule may attempt to serve class 2 customers before class 1 customers when there are no tardy jobs in queue, and hence will counteract the proposed due-date setting policies. The (PROP,SPT) due-date management policy was the only case where a priority sequencing rule had a significant impact under a particular due-date setting policy. The PROP due-date setting rule worked well with the SPT sequencing rule in this example because class 1 jobs have a shorter cycle time than class 2 jobs under the SPT rule.

We now turn our attention to Table II, which gives results for Problem II. The results for this case are quite similar to those of Problem I, but are even more dramatic: the best due-date setting policies cut the DDLT by a factor of two or three compared to conventional due-date setting policies. In contrast to Problem I, the non-parametric rules now out-performed the parametric rules. Also, there was virtually no difference in performance between the $E[S_{k,t}]$ and $[\sigma[S_{k,t}]$ rules in Problem II. HOT II outperformed NONPAR II, but the increase in performance was again modest. The service level target $\bar{\tau} = .5$ once again was within the 95% confidence interval of the observed average job tardiness for the (NONPAR II,SPT) and (HOT II,SPT) policies; however, the target was near the endpoint of the interval in both cases.

As in Problem I, there was very little difference among priority sequencing policies under our proposed due-date setting policies. Under the traditional due-date setting policies, SPT did not perform as well as the due-date sequencing policies. In particular, the (PROP,SPT) policy, which had performed well in Problem I, did not perform well in Problem II. Also, our study agrees with the results of Baker and Bertrand [3] in that the MDD rule performs better than the other sequencing rules in most cases.

In summary, all the due-date setting policies described in this paper easily outperformed the traditional due-date setting policies, and provided a much larger improvement in performance than did priority sequencing. Moreover, the proposed due-date setting

policies are very robust with respect to the sequencing policy that was used with it.

Acknowledgements

I am grateful to J. Michael Harrison for many helpful discussions about scheduling. I would also like to thank Dimitris Bertsimas and Stephen C. Graves for helpful comments.

REFERENCES

[1] Baker, K. R., "Sequencing Rules and Due-Date Assignments in a Job Shop," *Mgmt. Sci.* 30, 1093-1104 (1984).

[2] Baker, K. R. and J. W. M. Bertrand, "A Comparison of Due-Date Selection Rules," *AIIE Trans.* 13, 123-131 (1981).

[3] Baker, K. R. and J. W. M. Bertrand, "A Dynamic Priority Rule for Scheduling Against Due-Dates," *J. Ops. Mgmt.* 3, 37-42 (1982).

[4] Bertrand, J. W. M., "The Effect of Workload Dependent Due-Dates on Job Shop Performance," *Mgmt. Sci.* 29, 799-816 (1983).

[5] Bookbinder, J. H. and A. I. Noor, "Setting Job-Shop Due-Dates with Service Level Constraints," *J. Opl. Res. Soc.* 36, 1017-1026 (1985).

[6] Cobham, A., "Priority Assignment in Waiting Line Problems," *Ops. Res.* 2, 70-76; correction *ibid.* 3, 547 (1954).

[7] Conway, R. W., Maxwell, W. L. and Miller, L. W., *Theory of Scheduling*. Addison-Wesley, Reading, Mass. (1967).

[8] Dolan, R. J., "Incentive Mechanisms for Priority Queueing Problems," *Bell Journal of Economics* 9, 421-436 (1978).

[9] Eilon, S. and I. G. Chowdhury, "Due Dates in Job Shop Scheduling," *Intl. J. Prod. Res.* 14, 223-237 (1976).

[10] Gross, D. and C. M. Harris, *Fundamentals of Queueing Theory*, 2nd Edition. John Wiley and Sons, New York (1985).

[11] Harrison, J.M., "Dynamic Scheduling of a Multiclass Queue: Discount Optimality," *Ops. Rsch.* 23, 270-282 (1975).

[12] Harrison, J. M., "Brownian Models of Queueing Networks with Heterogeneous Customer Populations," in W. Fleming and P. L. Lions (eds.),

Stochastic Differential Systems, Stochastic Control Theory and Applications, IMA Volume 10, Springer-Verlag, New York, 147-186 (1988).

[13] Jagerman, D. L., "An Inversion Technique for the Laplace Transform with Application to Approximation," *Bell Syst. Tech. J.* 57, 669-710 (1978).

[14] Keilson, J., W. R. Nunn, and U. Sumita, "The Bilateral Laguerre Transform", *Appl. Math. Comput.* 8, 137-174 (1981).

[15] Kesten, H. and J. T. Runnenburg, "Priority Waiting Line Problems I,II," *Koninki. Ned. Akad. Wetenschap., Proc. Ser. A.* 60, 312-336 (1957).

[16] Kleinrock, L., *Queueing Systems Vol. I: Theory*. John Wiley and Sons, New York (1975).

[17] Klimov, G. P., "Time Sharing Service Systems I," *Th. Prob. Appl.* 19,532-551 (1974).

[18] Platzman, L., J. C. Ammons, and Bartholdi, J. J. III, "A Simple and Efficient Algorithm to Compute Tail Probabilities from Transforms," *Ops. Res.* 36, 137-144, (1988).

[19] Seidmann, A. and M. L. Smith, "Due Dates Assignment for Production Systems," *Mgmt. Sci.* 27, 571-581 (1981).

[20] Tcha, D. W. and S. R. Pliska, "Optimal Control of Single-Server Queueing Networks and Multi-Class M/G/1 Queues with Feedback," *Ops. Rsch.* 25, 248-258 (1977).

[21] Vepsalainen, A. P. J. and T. E. Morton, "Priority Rules for Job Shops with Weighted Tardiness Costs," *Mgmt. Sci.* 33, 1035-1047 (1987).

[22] Weeks, J. K., "A Simulation Study of Predictable Due-Dates," *Mgmt. Sci.*, 25, 4, 363-373 (1979).

[23] Wein, L. M., "Scheduling Semiconductor Wafer Fabrication," *IEEE Trans. Semiconductor Mfg.* 1, 115-130 (1988).

[24] Wein, L. M., "Optimal Control of a Two-Station Brownian Network," to appear in *Math. of Oper. Res.* (1988).

[25] Wein, L. M., "Scheduling Networks of Queues: Heavy Traffic Analysis of a Two-Station Network With Controllable Inputs," to appear in *Ops. Res.* (1988).

3-11-92